Paper AAS 01-467

American Astronautical Society

AIAA
American Institute of
Aeronautics and Astronautics

A Shape-Based Algorithm for the Automated Design of Low-Thrust, Gravity-Assist Trajectories

Anastassios E. Petropoulos[†]
Doctoral Candidate

James M. Longuski
Professor

School of Aeronautics and Astronautics,
Purdue University, West Lafayette, IN 47907-1282, USA.

[†] Currently, Senior Member of the Engineering Staff
Jet Propulsion Laboratory,
Pasadena, CA 91109-8099, USA.

AAS/AIAA Astrodynamics Specialists Conference

Québec City, Québec, Canada July 30 – August 2, 2001
AAS Publications Office, P.O. Box 28130, San Diego, CA 92198

AAS 01-467

A Shape-Based Algorithm for the Automated Design of Low-Thrust, Gravity-Assist Trajectories

Anastassios E. Petropoulos*and James M. Longuski†

School of Aeronautics and Astronautics, Purdue University,
West Lafayette, IN 47906-1282, USA.

Abstract

Given the benefits of coupling low-thrust propulsion with gravity assists, techniques for easily identifying candidate trajectories would be extremely useful to mission designers. We describe the computational implementation of an analytic, shape-based method for the design of low-thrust, gravity-assist trajectories. We also augment the method by allowing coast arcs to be patched with thrust arcs on the transfers between bodies. This approach permits not only rapid, broad searches over the design space, but also provides initial guesses for trajectory optimization. Numerical examples are presented for an Earth-Mars-Ceres rendezvous trajectory and an Earth-Venus-Earth-Mars-Jupiter flyby trajectory.

INTRODUCTION

It is well known that highly efficient, continuous-thrust propulsion and use of the gravity assist concept each provide significant benefits in trajectory design. Furthermore, each has been demonstrated in practice: NASA's *Deep Space 1* spacecraft is currently providing the first validation of interplanetary use of solar electric propulsion, while gravity assists have been repeatedly used in the exploration of the solar system, perhaps most notably by Voyager II launched in 1977. The coupling of high I_{sp}, low-thrust propulsion with gravity assists is a natural next step in the development of highly efficient trajectory design techniques for deep space missions. In the literature, the design of such trajectories is typically treated as an optimisation problem, which can be solved by a variety of techniques.[1-11] However, all techniques need some sort of initial guess for at least part of the solution, and even with such a guess, experience has shown that convergence to an optimal solution, particularly in the case of multiple gravity assists, is a formidable challenge. In this paper, rather than addressing the optimisation of particular initial guesses, we present a shape-based method[12,13] for efficiently generating such initial guesses for low-thrust, gravity-assist trajectories. These initial guesses serve a twofold purpose: They provide mission designers with rapid, broad overviews of the trajectory design space,[13] and they provide a starting point for trajectory optimisation.

We describe in this paper the computational method of solution of the equations resulting from the use of the exponential sinusoid shape[12,13] in a shape-based approach to trajectory design. We also discuss the incorporation of coast arcs into this model, and provide an example thereof. We demonstrate the efficacy of the shape-based approach in dealing with any number of flybys

*Doctoral Candidate; currently, Senior Member of the Engineering Staff, Jet Propulsion Laboratory, Pasadena, CA 91109-8099, USA; Member AAS and AIAA. Email: Anastassios.Petropoulos@jpl.nasa.gov

†Professor, Member AAS, Associate Fellow AIAA. Email: longuski@ecn.purdue.edu

with a sample trajectory from Earth to Jupiter using gravity assists at Venus, Earth and Mars. Furthermore, trajectories identified with the shape-based method are shown to be good initial guesses for optimisation.

In the shape-based approach, the spacecraft trajectory is assumed to be of a certain shape, with the requisite thrust profile determined therefrom. With the correct choice of shape, not only can we add to the small family of analytic solutions to the equations of motion (see for example Refs. 15–19), but can also obtain trajectories with satisfactory performance. In the present paper, we assume that the thrust arcs follow the planar exponential sinusoid shape. The out-of-plane motion required to encounter a gravity-assist body is assumed to be small and is approximated through an analysis of the orbital angular momentum vector. In previous research,[13] the planar motion the spacecraft was assumed to follow an exponential sinusoid between gravity-assist bodies. Here we add the capability of patching exponential sinusoid arcs with conic arcs, yielding thrust-coast or coast-thrust arcs between bodies. We provide the algorithmic details of how solutions are found for thrust, thrust-coast and coast-thrust legs. There are two key problems requiring numerical solution: 1) computation of the intersection points of two transcendental curves (namely, the trajectory arc and the target's orbit), and 2) subsequent solution of the targeting problem. We also incorporate the previously existing capability for purely ballistic transfer between bodies.[20]

METHODOLOGY

Overview

In this section, we describe how exponential-sinusoid-based thrust arcs are incorporated into the STOUR (Satellite Tour Design Program) software,[20] to form a new program, STOUR-LTGA, for the automated searching for low-thrust, gravity-assist trajectories. As in STOUR, for a given sequence of gravity-assist bodies, a range of launch dates and a range of launch v_∞s are automatically searched for trajectories, subject to various constraints, such as time of flight and propellant consumption constraints. In STOUR-LTGA, as in STOUR, the positions and velocities of the solar system bodies are modelled by polynomial representations, or, if the user so requests, by more accurate ephemeris data.

Previous papers[12,13] have presented an analysis of various thrust profiles that can be used to follow an exponential sinusoid shape. Of these, we choose the tangential thrust case, as it is the simplest analytic case, it is less prone to singularities, and it has tolerable thrust levels and attractive velocity profiles for both flyby missions and rendezvous missions. For convenience, in polar coordinates (r, θ) we repeat here the equations of motion, and the shape, flight path angle (γ), angular rate, and thrust equations, with the simplifications afforded by the tangential thrust assumption:

$$\begin{aligned}
\ddot{r} - r\dot{\theta}^2 + \frac{\mu}{r^2} &= F \sin\alpha \\
\frac{1}{r}\frac{d}{dt}\left(r^2\dot{\theta}\right) &= F\cos\alpha \\
r &= k_0 e^{k_1 \sin(k_2\theta + \phi)} && (1) \\
\tan\gamma &= k_1 k_2 c && (2) \\
\dot{\theta}^2 &= \left(\frac{\mu}{r^3}\right)\frac{1}{\tan^2\gamma + k_1 k_2^2 s + 1} && (3) \\
a &= \frac{(-1)^n \tan\gamma}{2\cos\gamma}\left[\frac{1}{\tan^2\gamma + k_1 k_2^2 s + 1} - \frac{k_2^2(1 - 2k_1 s)}{\left(\tan^2\gamma + k_1 k_2^2 s + 1\right)^2}\right] && (4)
\end{aligned}$$

where the dot, $(\dot{\ })$, denotes differentiation with respect to time, μ is the gravitational parameter of the central body, F is the magnitude of the thrust acceleration, α is the thrust angle, k_0, k_1, k_2 and ϕ are constants, a is the normalised thrust acceleration defined as

$$a \equiv \frac{F}{\mu/r^2} \tag{5}$$

$s \equiv \sin(k_2\theta + \phi)$, and $c \equiv \cos(k_2\theta + \phi)$. The thrust angle is given by

$$\alpha = \gamma + n\pi \tag{6}$$

where n is chosen so that the right-hand-side of Eq. 4 is positive.

The motion of the spacecraft between planets (*i.e.* on each leg) is permitted either to be purely conic, that is, coasting, or to involve some degree of thrusting. The purely conic legs are computed using the previously existing capabilities of STOUR. The remainder of this section describes how legs involving thrust are computed.

In STOUR-LTGA, legs can include thrust arcs in three ways: Either the entire leg is a thrust arc, or a thrust arc is succeeded or preceded by a coast arc. In other words, the legs may be described as thrust, thrust-coast, or coast-thrust. The motion on thrusting arcs is considered in two separate parts, the in-plane motion and the out-of-plane motion. The in-plane motion is assumed to follow a tangential-thrust exponential sinusoid (Eq. 1). The gravity assist is modelled as a hyperbolic-flyby-induced, instantaneous change in heliocentric spacecraft velocity without change in heliocentric position.

The out-of-plane motion is based on an analysis of the orbital angular momentum vector, where the out-of-plane angle and speed are approximated from the in-plane angular momentum and velocity components. The approximations are increasingly better, the smaller the out-of-plane angles.

In-Plane Motion

Thrust Legs

The STOUR-LTGA program steps through the launch dates and launch v_∞s specified by the user, and for each computes trajectories that reach the next body. The launch v_∞ vector is assumed to lie in the body's orbit plane, pointing in any direction. In the case of a flyby, a B-Plane angle[21] of 0° or 180° is assumed (the fundamental plane being taken as the flyby body's orbit plane), and the v_∞ turn angle is constrained by altitude. The low-thrust reference plane (*i.e.* where the planar motion occurs) for the next leg is taken as normal to the spacecraft's initial angular momentum vector. The program steps through the full range of in-plane orientations for the outgoing v_∞ vector. For each outgoing v_∞ vector, there correspond a heliocentric flight path angle and speed; meaning that $\tan\gamma$ and θ are known. Hence, the quantity

$$k_{12s} \equiv k_1 k_2^2 s \tag{7}$$

can be determined from Eq. 3. Then, together with the trigonometric identity $s^2 + c^2 \equiv 1$ and Eq. 2, there results the constraint relationship

$$k_1^2 k_2^4 - k_2^2 \tan^2\gamma - k_{12s}^2 = 0 \tag{8}$$

between the as yet undetermined shape paramters k_1 and k_2. Thus there remains one free shape parameter, taken as k_2, which is used to effect the targeting of the next body.

The search over k_2 for exponential sinusoids meeting the target body is narrowed in several ways. Without loss of generality, only positive values are considered for k_1 and k_2. To avoid singularities around periapsis, we require

$$1 - k_1 k_2^2 > 0 \tag{9}$$

which ensures that the denominator in Eq. 3 is always positive (when zero or negative, the exponential sinusoid cannot be followed using tangential thrust). For practical purposes, upper limits of one and two are imposed on k_2 and k_1, respectively, since the required thrust levels become untenably high when $(1 - k_1 k_2^2)$ approaches zero. However, in the case of k_1, the user may impose a different upper limit, typically a higher one, in order to allow greater excursions in radial distance on legs where this may be warranted. The lower limit on k_2 is taken as 0.01, as this permits up to 50 revolutions around the sun between periapsis and apoapsis — a number not likely to be exceeded in practice. Lastly, a range of k_2 values can be determined for which the resulting exponential sinusoid will intersect the projection onto the low-thrust reference plane of the target body's orbit. For outbound targets, *i.e.,* those whose minimum projected radius, $\rho_{\min}$, is greater than the current radius, r_B, we obtain, after some algebraic manipulations based on Eqs. 1 and 8

$$k_2^2 \leq \frac{\tan^2\gamma - 2k_{12s}\ln\left(\rho_{\min}/r_B\right)}{\left[\ln\left(\rho_{\min}/r_B\right)\right]^2} \tag{10}$$

For inbound targets, *i.e.,* those whose maximum projected radius, $\rho_{\max}$, is greater than the current radius, we similarly obtain

$$k_2^2 \leq \frac{\tan^2\gamma + 2k_{12s}\ln\left(r_B/\rho_{\max}\right)}{\left[\ln\left(r_B/\rho_{\max}\right)\right]^2} \tag{11}$$

For all other targets, no additional constraints can be imposed on k_2. In all cases, the limits on k_2 are adjusted, if necessary, based on the limits for k_1 and the constraint relationship Eq. 8.

Thus, for each turn angle we establish a range of k_2 values which yield intersections of the exponential sinusoid and the projected orbit of the target. All that remains is to solve for the specific value of k_2, if any, which yields the correct time of flight to the intersection. This is done by stepping through the range of k_2 values, and at each step computing the location of the intersection point. No analytic solution being available, this is computed using a step-size and search-direction controlled Newton method, with a suitable initial guess, as explained below. The time of flight to the intersection is then computed by quadrature, allowing a miss angle to be found — that is, the spacecraft-sun-target angle when the spacecraft reaches the intersection point (with the target projected onto the plane). Thus, as explained below, we search for a zero miss angle in the turn angle - k_2 space. There is a one-dimensional continuum of such solutions, which we sample at intervals according to the turn angle and k_2 step sizes. Of course, each exponential sinusoid will usually intersect the target's orbit more than once. We only keep track of up to two outbound intersections, and up to two inbound intersections.

Thrust-Coast and Coast-Thrust Legs

For thrust-coast or coast-thrust legs, which might be termed "mixed" legs, the point at which the thrust is turned off or on is called the switch point. The user specifies the heliocentric radial distance of the switch point (the switch radius), and may optionally specify whether the spacecraft is to be heliocentrically in- or out-bound at the switch point (the switch direction). The default switch direction is inbound for targets that are inbound from the switch radius, and outbound for outbound targets. If the switch radius is between the target's projected apoapsis and periapsis, then both switch directions are considered by default. Multiple coast revolutions around the sun may be optionally specified.

The search for encounters with the target body is accomplished in an analogous way to the thrust-only legs, using a turn angle - k_2 search grid. For the thrusting arcs of mixed legs, the constraint relation Eq. 8 is still valid, of course. In the case of thrust-coast legs, for each turn angle in the permitted range, the upper limit on k_2 is provided by Eq. 10 or Eq. 11, where the switch radius takes the place of ρ_{min} when the switch radius is greater than r_B, or takes the place of ρ_{max} when the switch radius is less than r_B. Intersection points and miss-angles are computed as with thrust legs, except that the exponential sinusoid is replaced by a conic in seeking the intersection with the projection of the target's orbit. In the case of coast-thrust legs, for each turn angle, the spacecraft coasts to the switch point, where its state may be easily calculated, yielding a flight path angle and speed which are used as in the thrust-only case for determining the upper limit on k_2. Intersection points and miss-angles are then computed exactly as with the thrust-only case.

Finding Intersections with the Target Body's Orbit

For each turn angle, the target body's orbit, assumed conic with semi-latus rectum p, and eccentricity e, is projected onto the low-thrust reference plane. The projected radius, ρ, after a short derivation, is found to be

$$\rho = \frac{p|\cos i|}{+\sqrt{1-\sin^2 i \cos^2\theta} \; + \; e\left[|\cos i| \cos\omega \cos\theta + \text{sgn}(\cos i)\sin\omega\sin\theta\right]} \tag{12}$$

where i is the inclination with respect to the low-thrust reference plane, θ is the polar angle in the low-thrust reference plane, and ω is the argument of periapsis. The angles θ and ω are measured from the positive x-axis, taken as the direction of the ascending node from the sun.

To find the intersection point(s), (r, θ), of the spacecraft trajectory with the target's projected orbit, Eq. 12 must be solved with the equation for the shape of the trajectory. The shape is either the exponential sinusoid (Eq. 1) for thrusting arcs, or the conic for coasting arcs. In both cases, the equations are transcendental in θ, with no analytic solution readily available. Thus, the intersection point must be found by a numerical root-finding technique. A step-size and step-direction controlled Newton method is used to solve for the roots of the difference, d_i, in the inverses of the radii:

$$d_i(\theta) \;=\; \frac{1}{\rho(\theta)} - \frac{1}{r(\theta)} \;=\; 0 \tag{13}$$

where $r(\theta)$ is the shape of the trajectory arc. The inverse radius is used in order to simplify the expression for the derivative required by the Newton method, thus speeding up computations, convergence rates aside.

The root finding method is best described by means of an example. We consider the intersection of an exponential sinusoid in the ecliptic plane, having periapsis at 0.8 AU, apoapsis at 3.1 AU, and $k_2 = 0.1$, with the projected orbit of the asteroid Ceres (semi-major axis 2.77 AU, eccentricity 0.077, and inclination 10.6°). For the exponential sinusoid and the projected orbit, radius as a function of θ is shown in Fig. 1. The difference in inverse radius d_i is shown as a function of θ in Fig. 2. Let us assume that the spacecraft's initial position is below Ceres' orbit (*i.e.*, Ceres is outbound from the spacecraft). The sequential intersection points with Ceres' projected orbit are labelled as B, C, D, G in the two figures. The goal is to find up to two outbound and two inbound intersections, namely the points B, C, and G, respectively (there is only one inbound intersection).

The upper and lower bounds on the projected radius, ρ_{x} and ρ_{n}, are conservatively estimated as the apoapsis and periapsis radii of Ceres, respectively, multiplied by $|\cos i|$. For the first outbound intersection (point B), the initial guess for the value of θ solving Eq. 13 is taken as θ_A, the easily determined value of θ where the exponential sinusoid radius is equal to ρ_{n}. We note that the root is guaranteed to be beyond θ_A. The Newton method will then quickly converge to θ_B, without any need for exercising step control. For the first inbound intersection (point G), the point F, with polar angle θ_F, is used as the initial guess. (If the exponential sinusoid's apoapsis radius were below

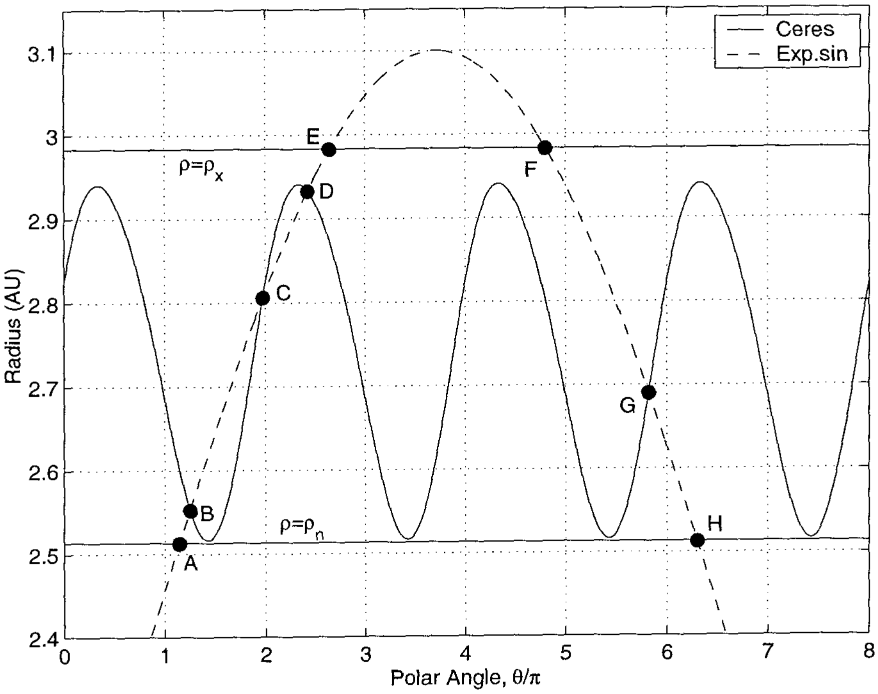

Figure 1 Intersections of an exponential sinusoid arc in the ecliptic with the projection of Ceres' orbit.

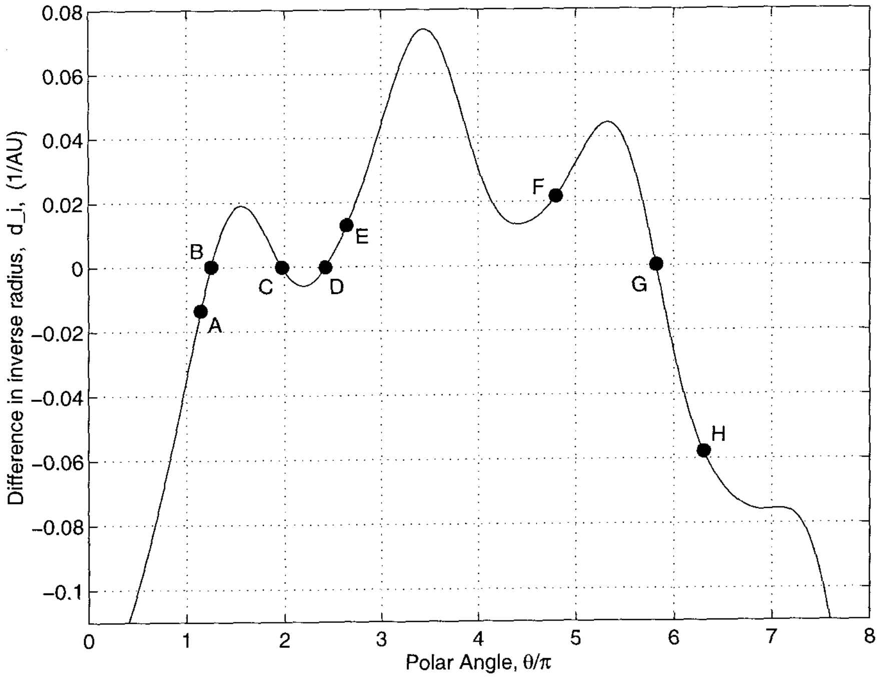

Figure 2 Root-finding for the intersections of an exponential sinusoid with the projection of a conic section.

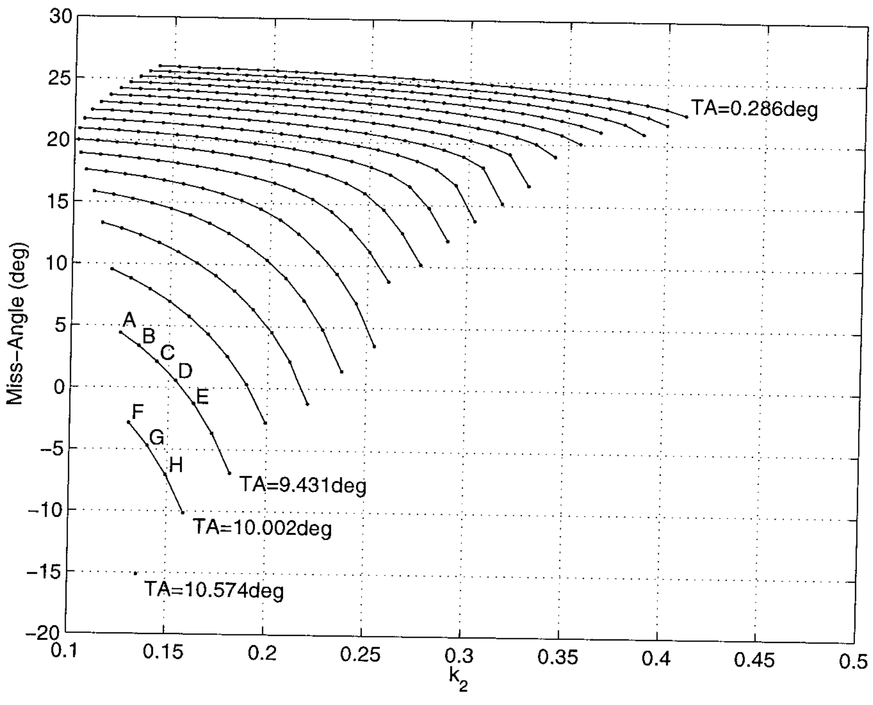

Figure 3 Miss-angle as a function of k_2 with turn-angle (TA) contours at 0.572° increments for exponential sinusoid arcs originating at Mars and intersecting Ceres' projected orbit.

ρ_X, then the θ of apoapsis would be used as the initial guess.) Clearly, the first step would take us away from G, and so step control is invoked, setting the next initial guess just over $\frac{\pi}{2}$ beyond θ_F. Decreases in θ are only allowed after the first step, and then only if a root of d_i is straddled. The largest permitted decrease is just under $\frac{\pi}{2}$. Thus, convergence to the first outbound and the first inbound intersections is relatively robust.

The second outbound intersection (point C) is computed with lesser certainty of convergence. The initial guess is taken as the lesser of $(\theta_B + 0.1\,\text{rad})$ and $(4\theta_B + \theta_G)/5$. Except in rare circumstances, the former value will be normally be the lesser of the two. The same step controls as for the first intersections are still imposed. With these step controls and initial guess, if the point C were too close to point B, it would be skipped, with the algorithm converging to the next intersection (point D) if any. In a sense, this weeds out intersections that have similar characteristics due to proximity. The second inbound intersection, which in the depicted case does not exist, would be found by using $(\theta_G + 0.1\,\text{rad})$ as the initial guess. The sampling of up to a total of four intersections, with their differing characteristics, is deemed to be sufficiently representative of the available solutions.

While the above root-finding example is for an outbound target body, the root-finding for inbound targets is entirely analogous. Also, the case where the spacecraft is on a conic arc, rather than an exponential sinusoid arc, is treated similarly, although the problem is somewhat simpler, as the angular period of the spacecraft is equal to the angular period of the target body (2π), whereas on an exponential sinusoid, the spacecraft's angular period can be significantly greater than the target's.

Grid Search for Encounters with the Target Body

Once the intersection points between the trajectory and the target's orbit have been computed for a given turn angle (TA) and k_2, the time of flight to these intersections is computed by quadrature. Thus, when the spacecraft arrives at an intersection, the projected position of the target body is known, making it possible to compute the angle subtended at the sun by the spacecraft and the projected position. This miss-angle is taken as positive when the target leads the spacecraft, negative when it lags. Clearly, an encounter occurs when the miss-angle is zero.

The process by which encounters are found may be depicted graphically, by plotting the miss-angle versus k_2, with contours for different turn angles. Of course, the miss-angles on any one plot should come from the same type of intersection, say the first outbound intersection, or the first inbound intersection. Fig. 3 shows such a plot for the first outbound intersection of a Mars-Ceres leg of an Earth-Mars-Ceres trajectory with thrust-only legs. Only a representative sample of TA contours is shown.

The miss-angles are computed for points on each TA contour in turn, starting at the lower end of the available k_2 range for the contour. Extrapolation for zero-miss-angle is done both across TA contours and along TA contours. If two points on a contour straddle a zero-miss-angle (such as D and E in Fig. 3), then a linear extrapolation is made over k_2. The miss-angle for the extrapolated value is then computed; it is retained if it falls within a certain tolerance of zero. When the next TA contour is computed, extrapolations are made between it and the previous contour, if warranted. For example, the point F is closest in k_2 value to point B on the previous TA contour, and the two points straddle a zero-miss-angle, warranting extrapolation. First a linearly extrapolated value is obtained for the turn angle, ignoring differences in k_2. The second linear extrapolation is between the k_2 values of points B and F, but for the extrapolated turn angle value. The actual miss-angle for the extrapolated turn angle and k_2 is again checked against the tolerance. The remaining points on the contours are similarly evaluated for root straddling.

To reduce the computational memory requirements, at any one time, information is only stored for up to two TA contours. For example, the third TA contour will overwrite the first, as the first is no longer needed. We note that separate miss-angle calculations must be made for the separate cases of different intersections, different switch directions, and different full coast revolutions.

Out-of-plane motion

Two cases must be distinguished. The spacecraft may encounter the target body whilst either on a thrust arc or on a coast arc. In both cases, the target's out-of-plane position at the time of the in-plane encounter is matched by an additional thrust acceleration, f_h, acting along or against the spacecraft's angular momentum vector for some final portion of the leg's thrust arc. This out-of-plane thrust is assumed to have the form

$$f_h = \left(a_{0P} + b_0 \frac{r_{\min}}{r}\right) \frac{\mu}{r^2} \tag{14}$$

where a_{0P} and b_0 are constants, and $r_{\min}$ is the periapsis radius of the exponential sinusoid being followed. The positive f_h direction is taken to be along the angular momentum vector.

Since both the in- and out-of-plane thrust will be much smaller than the gravitational attraction of the central body, the in-plane components of the specific angular momentum, h_x and h_y, behave according to

$$\frac{\mathrm{d}h_x}{\mathrm{d}\theta} \approx \frac{r f_h \sin\theta}{\dot{\theta}} \tag{15}$$

$$\frac{\mathrm{d}h_y}{\mathrm{d}\theta} \approx -\frac{r f_h \cos\theta}{\dot{\theta}} \tag{16}$$

where the x direction is taken as lying along $\theta = 0$. We note that to first order, the in-plane thrust does not affect the in-plane angular momentum components. Now, h_x and h_y remain small, so that the total angular momentum is approximately equal to just the out-of-plane component, $r^2\dot{\theta}$, computed as if f_h were zero. Thus, at any point on an out-of-plane thrust arc, the out-of-plane angle, ϕ, and the speed normal to the plane, v_z, can be approximated as

$$\begin{aligned} \tan\phi &\approx -\frac{h_x \cos\theta + h_y \sin\theta}{r^2\dot{\theta}} \qquad (17) \\ v_z &\approx \left[h_x(\sin\theta - \tan\gamma\cos\theta) - h_y(\cos\theta + \tan\gamma\sin\theta)\right]/r \qquad (18) \end{aligned}$$

For the thrust-to-encounter case, only the target's out-of-plane position is matched, meaning that only Eq. 17 need be satisfied, and not Eq. 18. Setting $b_0 = 0$ and keeping only the unknown constant a_{0P} in the expression f_h is sufficient to do so uniquely. For the coast-to-encounter case, both an out-of-plane speed and position must be matched at the end of the preceding thrust arc, if the spacecraft is to match the target's out-of-plane position at the end of its coast arc. Thus, in this case, both a_{0P} and b_0 are needed in the expression for f_h, as two equations must be satisfied, Eqs. 17 and 18.

Since the integrals, h_x and h_y, in Eqs. 17 and 18 must be computed numerically, they are evaluated backwards from the in-plane intersection with the target, to permit, at each integration step, the evaluation of the required f_h. The "best" f_h is the one used. In the case of thrust-to-encounter this means the f_h with the lowest a_{0P}. For the coast-to-encounter case, this means the f_h with the lowest average value of $|f_h r^2/\mu|$, with the average computed over θ.

The effect of the out-of-plane motion on the time of flight is ignored. This approximate method permits rapid computations, and is increasingly accurate for smaller out-of-plane excursions. While this method is incorporated into STOUR-LTGA, we generally give little consideration to the out-of-plane thrust and associated propellant, since, other than a simple selection process for the out-of-plane thrust, this method was developed with regard neither for attaining the most satisfactory thrust profile nor for the benefits to be had by using different B-plane angles.

Propellant Consumption

An estimate of the propellant consumption is made by assuming a constant specific impulse (I_{sp}) for the low-thrust engines. This simplification permits the required propellant mass to be expressed as a fraction of the initial spacecraft mass. In addition, we compute separately the propellant fractions required for the in-plane and out-of-plane thrust; from these is then computed a single fraction. Since our approach controls directly only the shape of the trajectory, the required thrust may fall outside the range of more accurate thruster models, such as those used by Sauer.[10] This fact, together with the additional computational burden required by the more accurate thruster models, prompt the constant I_{sp} assumption. The more sophisticated models can be employed in the optimisation of specific trajectories selected from the broad searches permitted by our method.

Automated Leg Selection and Recording of results

At any body, with a given incoming trajectory, there exists for the next leg a whole continuum of trajectory solutions in the turn-angle - k_2 space. The computational approach samples this continuum at intervals. From each of these sampled solutions, there again arises a continuum of solutions to the next body. Thus, we limit the selection of solutions based on various criteria, to avoid an exponential increase in the number of trajectories that reach the final body. Solutions for a given leg are stored and propagated if, for example, they have the lowest time of flight, the lowest propellant mass fraction, the lowest thrust levels, the lowest arrival v_∞, the highest arrival

v_∞, and so on. Currently, the program uses eighteen criteria, which can be activated or deactivated individually for each leg.

Any trajectories found by STOUR-LTGA are printed chronologically by launch date to a computer file with a one- or two-line listing for each leg of the trajectory. The user may request that partial trajectories be printed also. In this case, for launch date and launch v_∞ values where the final body in the path could not be reached, leg data is printed up to the last body reached.

STOUR-LTGA can also divide the legs into isochronous segments (the user can select how many for each leg), and list the $\Delta\bar{V}$ accumulated on each segment due to the thrust, if any. Such output is useful as input for trajectory optimisation software that uses direct methods.

RESULTS

The new STOUR-LTGA program, described in the previous section, can be applied[22] to a variety of mission design problems. As representative missions, we present here a rendezvous mission from Earth (E) via Mars (M) to the asteroid Ceres (C), and a flyby mission from Earth to Jupiter (J) after flybys of Venus (V), Earth and Mars. For ease of reference, Table 1 lists the semimajor axis, eccentricity, and inclination for the heliocentric orbit of each body.

An optimised instance of the EMC mission type has been reported in the literature by Sauer.[10] The EVEMJ mission type has not been found in the literature. Indeed, nor has any three-gravity-assist, low-thrust trajectory appeared from a perusal of the literature. Selected trajectories from the broad searches conducted with STOUR-LTGA are used as initial guesses in optimisation. The optimiser, intended for preliminary optimisation of trajectories, uses the direct method described by Sims and Flanagan,[11] who most generously made available their prototype program, which was subsequently rewritten T. Troy McConaghy. The rewritten optimiser currently has the name GALLOP (Gravity-Assist, Low-thrust, Local Optimisation Program).[23]

Table 1

ORBITAL ELEMENTS FOR SELECTED BODIES

Body	a (AU)	e	i
Venus	0.72	0.007	3.4°
Earth	1.00	0.017	0.0°
Mars	1.52	0.093	1.8°
Jupiter	5.20	0.049	1.3°
Ceres	2.77	0.077	10.6°

Earth - Mars - Ceres

A broad search over the launch years 1990 to 2049, with launch v_∞s between 0.75 km/s and 2.00 km/s, reveals that the year 2003, the month of April in particular, has one of the higher concentrations of low arrival v_∞ trajectories. Both legs are assumed to be thrust-only. Fig. 4 shows a scatter plot of the ecliptic arrival v_∞ against the launch date. The April launch date is close to that of an EMC rendezvous trajectory optimised by Sauer.[10] Thus, a more focused search is performed over the months of April and May 2003, with launch v_∞s between 1.0 km/s and 1.7 km/s. The arrival v_∞s that are below 0.1 km/s in Fig. 4 correspond to very high flight times. Thus, from the refined search, a better all-round trajectory is picked, with pertinent data shown in Table 2.

The selected trajectory from STOUR-LTGA was used as an initial guess in optimisation using GALLOP. The optimisation sought to maximise final spacecraft mass at the rendezvous with Ceres,

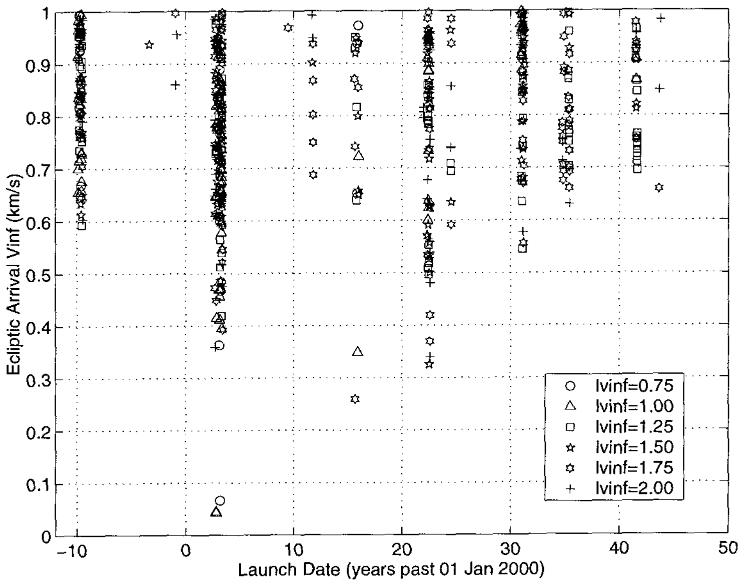

Figure 4 Ecliptic arrival v_∞s below 0.75 km/s for Earth-Mars-Ceres trajectories launching in April and May 2003.

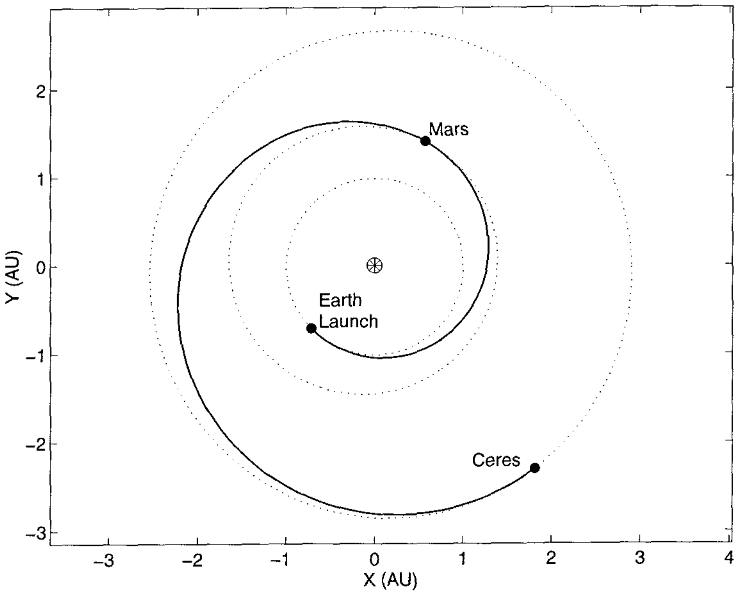

Figure 5 Ecliptic projection of an STOUR-LTGA Earth-Mars-Ceres trajectory.

Table 2

AN STOUR-LTGA EARTH-MARS-CERES TRAJECTORY, OPTIMISED IN GALLOP, AND COMPARED WITH AN OPTIMISED CASE BY SAUER

	STOUR-LTGA	GALLOP	Sauer[10]
Launch Date (dd/mm/2003)	06/05	06/05	08/05
Launch v_∞ (km/s)	1.60	1.60	1.37
Ecliptic Arrival v_∞ (km/s)	0.24	0.00	0.00
fmax, EM (mm/s^2)	0.18		
fave, EM (mm/s^2)	0.13		
fmax, MC (mm/s^2)	0.12		
fave, MC (mm/s^2)	0.08		
pmf	0.34[a]	0.23[b]	0.28[b]
TOF EM (days)	271	271	250
TOF MC (days)	862	862	845
TOF total (days)	1133	1133	1095

[a] Includes in-plane components only.

[b] Includes in- and out-of-plane components.

while keeping launch v_∞, and launch, arrival and flyby dates fixed. The GALLOP data and available data for the Sauer trajectory are also presented in the table. Both the GALLOP trajectory and the Sauer trajectory assume a single NSTAR ion thruster. The reference solar array power (*i.e.*, roughly the solar array power available at 1 AU) was 10 kW for GALLOP, and 5 kW for Sauer, who also permitted 125 W to be used for spacecraft housekeeping. The higher power used in GALLOP permits the ion engine to be used at full throttle (2.7 kW) until about 1.9 AU, whereas at 5 kW this distance decreases to about 1.4 AU. Of note is that the I_{sp} drops-off at lower throttle levels. Thus, the GALLOP trajectory reports a slightly lower propellant mass fraction than Sauer's case. Sauer assumes an initial spacecraft mass of 568 kg, which is also used by GALLOP.

The STOUR-LTGA trajectory in Table 2 shows good correspondence to Sauer's optimised trajectory. The trajectory geometry, shown in Fig. 5, is also similar: For example, the exponential sinusoid solution performs 0.56 revolutions about the sun on the EM leg and 0.67 revolutions on the MC leg, as compared with about 0.53 and 0.68 revolutions, respectively, for the optimised solution. Furthermore, the STOUR-LTGA trajectory provides a good starting guess in optimisation. Using GALLOP, we see rapid convergence to a solution that corresponds well to the Sauer solution which was optimised by a different means (indirect methods). We expect STOUR-LTGA to provide similarly close-to-optimal solutions not only on other launch dates for the EMC case, but also for Mars gravity assist trajectories to other asteroids.

Earth - Venus - Earth - Mars - Jupiter

In the literature, the optimal trajectories we have found have at most two flybys on thrusting legs between departure and destination planets. STOUR-LTGA, in addition to the strength of broad searching over launch date and v_∞, offers the significant ability of searching over different, long sequences of flyby bodies. One such path is EVEMJ (including launch and destination bodies). (Currently, paths of up to 20 bodies are permitted, although this limit can be easily increased. Of course, the longer the paths, the longer the computation times, and the longer the flight times, making the longer paths better suited for short-period bodies.)

For the EVEMJ path, where a flyby of Jupiter is desired, the launch years 1975 to 2049 are searched, for launch v_∞s of 0.5 km/s to 2 km/s at 0.5 km/s increments. These launch v_∞s are well shy of even the Hohmann launch v_∞ to Venus (2.5 km/s). Mars-Jupiter is assumed to be a thrust-coast leg, with a switch radius of 3 AU. The other three legs are assumed to be thrust-only

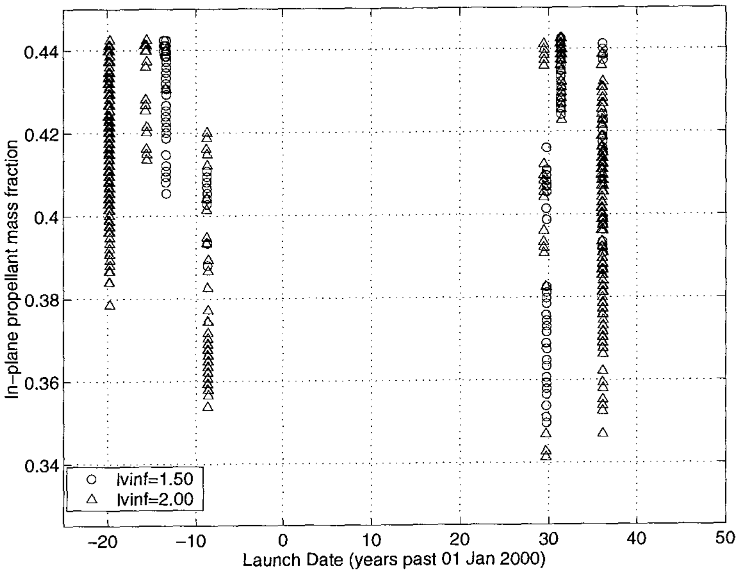

Figure 6 In-plane propellant consumption for EVEMJ trajectories.

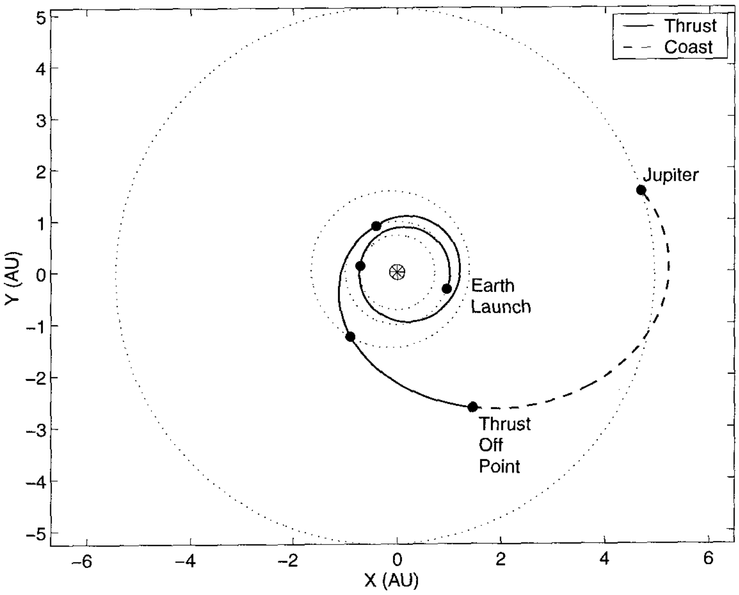

Figure 7 Ecliptic projection of an STOUR-LTGA EVEMJ trajectory.

Table 3

AN STOUR-LTGA EVEMJ TRAJECTORY AND ITS GALLOP-OPTIMISED VERSION

	STOUR-LTGA	GALLOP
Launch Date (dd/mm/yyyy)	03/09/2029	03/09/2029
Launch v_∞ (km/s)	2.00	2.00
V Flyby v_∞ (km)	3.64	4.30
V Flyby Altitude (km)	6533	20365
V Flyby B-Plane Angle[a]	-0.5°	62.7°
E Flyby v_∞ (km)	6.50	5.90
E Flyby Altitude (km)	655	763
E Flyby B-Plane Angle[a]	-166.0°	-175.0°
M Flyby v_∞ (km)	13.70	11.29
M Flyby Altitude (km)	200	200
M Flyby B-Plane Angle[a]	21.8°	-5.4°
J Flyby v_∞ (km/s)	5.85	6.25
fave, EV (mm/s^2)	0.12	
fave, VE (mm/s^2)	0.16	
fave, EM (mm/s^2)	0.10	
fave, MJ (mm/s^2)	0.14	
Switch radius (AU)	3.0	
pmf	0.341[b]	0.268[c]
TOF EV (days)	165	165
TOF VE (days)	334	314
TOF EM (days)	131	151
TOF MJ (days)	1335	1335
TOF Total (days)	1965	1965

[a] Fundamental plane taken as planet's equatorial plane.
[b] Includes in-plane components only.
[c] Includes in- and out-of-plane components.

legs. To keep the search times short, a launch date step of 10 days is used. Since the last leg, Mars-Jupiter, will be somewhat energetic, the upper limit on k_1 is set to 3. A maximum total time of flight of 2500 days is allowed, with the maximum leg flight times set at 500 days for each leg except the Mars-Jupiter leg, which is allowed up to 2000 days. The minimum flyby altitudes are all 200 km. Finally, with an I_{sp} of 2500 seconds assumed, conservative caps of 0.442, 0.25 mm/s^2, and 0.35 mm/s^2 are set on the in-plane propellant mass fraction, and the average and maximum thrust accelerations, respectively.

The broad search yields two main groupings of trajectories, spaced about 49 years apart, as seen in Fig. 6, where the in-plane propellant mass fraction is plotted against launch date. The 49-year spacing corresponds roughly with the 45-year repeat cycle noted in previous work[24] for the non-low-thrust E-VMVE-J path which utilises the same flyby bodies. Trajectories are found only for launch v_∞s of 1.5 km/s and 2 km/s, with the latter numbering about four times more than the former. Higher launch v_∞s and less restrictive constraints on the thrust levels would fill out the trajectory families and produce new ones.

We select from the broad search the trajectory with the lowest in-plane propellant mass fraction, which launches in 2029, as it also has attractive thrust and flight time characteristics when compared to the other trajectories in the 2029-2036 trajectory grouping. The trajectory geometry is depicted in Fig. 7, and pertinent data are listed in Table 3. The trajectory also serves as an initial guess in optimisation using GALLOP, where the maximum final mass is sought for a launch mass of 300 kg, a single NSTAR thruster is used, and a reference array power of 10kW is assumed. The launch v_∞, launch date, flyby and arrival dates are all held fixed. The optimal trajectory found by GALLOP is also reported in Table 3.

Comparing the unoptimised and optimised trajectories in Table 3, three main differences are seen. First, the B-plane angles are different, significantly so in the case of Venus, meaning that the optimiser was able to alter these to effect the out-of-plane targeting more efficiently than using thrust alone. Second, the flyby altitude at Venus is about three times larger in the optimised case. Third, the flyby date of Earth in the optimised case is 20 days prior to that in the unoptimised case. The last two differences are present largely as a way of reducing the exaggerated, lengthy, and wastefully thrusting excursion beyond 1 AU, exhibited by the unoptimised trajectory prior to the Earth flyby (see Fig. 7). One potential method of diminishing this excursion in STOUR-LTGA would be to use a thrust-coast leg for the VE transfer. As for the considerably higher, optimised Venus flyby altitude, it still causes the incoming v_∞ vector to be turned by 47°, corresponding to a heliocentric ΔV of 3.44 km/s. It is interesting to note that GALLOP did not converge as easily on this EVEMJ trajectory as it did on the EMC trajectory (see Table 2). Perhaps this is due to the added complexity of not only imposing two more flyby constraints, but also utilising the flybys most effectively . Lastly, both the optimised and unoptimised versions of this EVEMJ trajectory are seen to be comparable to, if not superior to, non-low-thrust, gravity-assist Earth-Jupiter trajectories reported previously.[24]

Other paths

When using STOUR-LTGA, the low-thrust, gravity-assist mission designer is faced with yet another choice — which trajectory path to use. We have seen a simple path and one more-complex path above. Other complex paths of similar character, such as EVMVEJ, can be easily identified. One class of transfers deserving special mention is that of consecutive flybys of the same body for v_∞-leveraging purposes. Whether interior or exterior leveraging is sought, the transfers should be set as coast-thrust in STOUR-LTGA, as can be explained by considering a tangential launch from Earth. In both the interior and exterior cases, a thrust-coast leg will not permit a reencounter of the Earth, for most reasonable values of the k_1 and k_2 parameters. In the interior case, the apoapsis of the trajectory's osculating ellipse is reduced below 1 AU, while in the exterior case, the periapsis is raised above 1 AU, making it impossible to return to the Earth on a coast arc. Using a thrust-only leg to return to the Earth would be possible, but pointlessly wasteful, as the arrival v_∞ at the Earth encounter would be the same as the launch v_∞ (assuming a circular Earth orbit), thanks to the symmetry of the exponential sinusoid. The coast-thrust leg does not have these difficulties. By extension, any transfer back to the same body, whether at launch or later in a path, should be set as a coast-thrust transfer.

CONCLUSIONS

From an infinity of solutions for any given low-thrust, gravity-assist trajectory type, how do we choose even just one, let alone one that is "best" in some sense? This thesis has presented a method for finding "sufficiently good" solutions from the infinity of solutions made possible by the availability of continuous thrust. We see that the shape-based approach, in particular using the exponential sinusoid shape, significantly simplifies the problem for low- to medium-energy trajectories. Broad searches are made possible over what are perhaps the six main items of interest to a mission designer: Launch date, launch v_∞, propellant mass, thrust levels, time of flight and arrival v_∞. An automated search capability for low-thrust, gravity-assist trajectories is added to the STOUR program. STOUR-LTGA can thus compute trajectories comprising legs that are thrust-only, coast-only, thrust-coast or coast-thrust. The method not only provides broad overviews of the trajectory design space, but also provides good starting points for trajectory optimisation.

ACKNOWLEDGEMENTS

This research has been supported in part by the Jet Propulsion Laboratory, California Institute of Technology under Contract Numbers 961211 and 1211514. We are grateful to Dennis V. Byrnes (Technical Manager) and Jon A. Sims for providing useful information, guidance and helpful suggestions. We also thank Jon Sims and Steve Flanagan for providing their direct-optimisation computer code, and T. Troy Mcconaghy for rewritting it and providing us with the new code.

REFERENCES

1. Kawaguchi, J., Takiura, K., and Matsuo, H., "On the Optimization and Application of Electric Propulsion to Mars and Sample and Return Mission," *Advances in the Astronautical Sciences,* Univelt Inc., San Diego, CA, Vol. 87, No. 1, 1994, pp. 539-556.
2. Thorne, J. D. and Hall, C. D., "Minimum-Time Continuous-Thrust Orbit Transfers," *Journal of the Astronautical Sciences,* Vol. 45, No. 4, Oct.-Dec. 1997, pp. 411-432.
3. Kluever, C. A., "Optimal Low-Thrust Interplanetary Trajectories by Direct Method Techniques," *Journal of the Astronautical Sciences,* Vol. 45, No. 3, July-Sept. 1997, pp. 247-262.
4. Kechichian, J. A., "Low-Thrust Trajectory Optimization Based on Epoch Eccentric Longitude Formulation," *Advances in the Astronautical Sciences,* Univelt Inc., San Diego, CA, Vol. 87, No. 2, 1994, pp. 863-884.
5. Coverstone-Carroll, V. and Williams, S. N., "Optimal Low Thrust Trajectories Using Differential Inclusion Concepts," *Journal of the Astronautical Sciences,* Vol. 42, No. 4, Oct.-Dec. 1994, pp. 379-393.
6. Betts, J. T., "Optimal Interplanetary Orbit Transfers by Direct Transcription," *Journal of the Astronautical Sciences,* Vol. 42, No. 3, July-Sept. 1994, pp. 247-268.
7. Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics,* Vol. 21, No. 2, March-April 1998, pp. 193-207.
8. Betts, J. T. and Huffman, W. P., "Sparse Optimal Control Software SOCS," Boeing Information and Support Services, Mathematics and Engineering Analysis Technical Document MEA-LR-085, The Boeing Co., Seattle, WA, July 1997.
9. Hui, Y. and Hongxin, W., "Initial Adjoint Variable Guess Technique and its Application in Optimal Orbital Transfer," AIAA/AAS Astrodynamics Specialist Conference, AIAA Paper 98-4551, Boston, MA, Aug. 10-12, 1998.
10. Sauer, C. G., "Solar Electric Performance for Medlite and Delta Class Planetary Missions," AAS/AIAA Astrodynamics Specialist Conference, AAS Paper 97-726, Sun Valley, Idaho, Aug. 4-7, 1997.
11. Sims, J. A. and Flanagan, S. N., "Preliminary Design of Low-Thrust Interplanetary Missions," AAS/AIAA Astrodynamics Specialist Conference, AAS Paper 99-338, Girdwood, Alaska, Aug. 1999.
12. Petropoulos, A. E., Longuski, J. M., and Vinh, N. X., "Shape-Based Analytic Representations of Low-Thrust Trajectories for Gravity-Assist Applications," AAS/AIAA Astrodynamics Specialist Conference, AAS Paper 99-337, Girdwood, Alaska, Aug. 1999.
13. Petropoulos, A. E. and Longuski, J. M., "Automated Design of Low-Thrust Gravity-Assist Trajectories," AIAA/AAS Astrodynamics Specialist Conference, AIAA Paper 2000-4033, Denver, Colorado, Aug. 2000.
14. Tsu, T. C., "Interplanetary Travel by Solar Sail," *J. American Rocket Society,* Vol.29, 1959, pp.422–427.
15. Lawden, D. F., *Optimal Trajectories for Space Navigation,* Butterworths, London, 1963.
16. Battin, R. H., *An Introduction to the Mathematics and Methods of Astrodynamics,* 1st ed. 4th printing, AIAA, New York, 1987.
17. Prussing, J. E. and Coverstone-Carroll, V., "Constant radial thrust acceleration redux," *J. Guidance, Control, and Dynamics,* Vol.21, No.3, May-June 1998, pp.516–518.
18. Markopoulos, N., "Non-Keplerian Manifestations of the Keplerian Trajectory Equation and a Theory of Orbital Motion Under Continuous Thrust," AAS/AIAA Space Flight Mechanics Meeting, AAS Paper 95-217, Albuquerque, New Mexico, Feb. 1995.
19. Bishop, R. H. and Azimov, D. M., "New Analytic Solutions to the Fuel-Optimal Orbital Transfer Problem Using Low-Thrust Exhaust-Modulated Propulsion," AAS/AIAA Space Flight Mechanics Meeting, AAS Paper 00-131, Clearwater, Florida, Jan. 2000.
20. Williams, S. N., "Automated Design of Multiple Encounter Gravity-Assist Trajectories," M.S. Thesis, School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN, Aug. 1990.

21. Sergeyevsky, A.B. and Snyder, G.C., "Interplanetary Mission Design Handbook, Volume I, Part 3," JPL Publication 82-43, Jet Propulsion Laboratory, Pasadena, CA, Dec. 1, 1982.

22. Petropoulos, A. E., "A Shape-Based Approach to Automated, Low-Thrust, Gravity-Assist Trajectory Design," Ph.D. Thesis, School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN, May 2001..

23. McConaghy, T. T., Debban, T., Petropoulos, A. E., and Longuski, J. M., "An Approach to Design and Optimization of Low-Thrust Trajectories with Gravity Assists," AAS/AIAA Astrodynamics Specialists Conference, Extended Abstract submitted, Québec City, Québec, Canada, July/Aug. 2001.

24. Petropoulos, A.E., Longuski, J.M., and Bonfiglio, E.P., "Trajectories to Jupiter via Gravity Assists from Venus, Earth, and Mars," *Journal of Spacecraft and Rockets*, Vol.37, No.6, Nov.-Dec. 2000, pp.776–783.